Forum Statistika dan Komputasi, April 2011 p : 35-43 Vol 16 No.1
ISSN : 0853-811

# PENERAPAN METODE *RANDOM FOREST* DALAM *DRIVER ANALYSIS*

(*The Application of Random Forest in Driver Analysis*)

**Nariswari Karina Dewi[1] , Utami Dyah Syafitri[2], Soni Yadi Mulyadi[3]**
[1] Mahasiswa Departemen Statistika, FMIPA-IPB
[2]Departemen Statistika, FMIPA-IPB
[3]PT. Ipsos Indonesia

E-mail : [1]naris.ayes@yahoo.com

***Abstract***

*Driver analysis is one approach to know which the greatest expalanatory variables influence the response variable. This analysis is well known in marketing research. In this area, explanatatory variables (X) and response variable (Y) ussually are measured by ordinal data and the relationship between those variables is non linier. One of the approach to build model on that situation is random forest. Two important things in random forest are size of random forest and sample size of X. In this research, we worked with simulation to know the size of random forest which give higher accuration and more stabil. The simulation showed that the best condition achieved when the size of random forest is 500 and the sample size of X is 4.*

*Key words* **:** *driver analysis, random forest, variable importance.*

## PENDAHULUAN

Persaingan pasar mendorong produsen untuk selalu memperbaiki kinerja produknya, misalnya kesediaan seseorang untuk membeli produk tersebut. Perbaikan dapat dilakukan secara efektif dan efisien jika diketahui prioritas atribut produk yang menggerakkan kinerja yang dimaksud. Dalam riset pemasaran, analisis yang digunakan untuk menghasilkan informasi tersebut dikenal dengan nama *driver analysis*.

*Driver analysis* didasarkan pada metode yang mengeksplorasi hubungan antara peubah penjelas dan peubah respons. Metode yang biasa digunakan antara lain yaitu analisis regresi dan analisis korelasi. Sementara itu, data yang dianalisis umumnya berupa data kategorik serta memiliki hubungan non-linier antara peubah penjelas dan peubah responsnya. Oleh sebab itu, diperlukan metode yang lebih sesuai dengan kondisi data. Salah satu metode tersebut adalah metode *random forest*.

*Random forest* didasarkan pada teknik pohon keputusan sehingga mampu mengatasi masalah non-linier. Metode ini merupakan metode pohon gabungan. Untuk mengidentifikasi peubah penjelas yang relevan dengan peubah respons, *random forest* menghasilkan ukuran tingkat kepentingan (*variable importance*) peubah penjelas. Dalam bidang biostatistika, hal tersebut diterapkan pada masalah *gene selection* pada data *microarray* (Díaz-Uriarte & Andrés 2006). Penerapan *random forest* dalam bidang biostatistika memang telah populer. Prioritas peubah penjelas dapat diketahui melalui ukuran tingkat kepentingan peubah penjelas. Oleh karena itu, metode *random forest* dapat diterapkan pada *driver analysis*. Penelitian ini mengkaji hal tersebut. Pada penelitian ini, *driver analysis* dilakukan dalam rangka memperbaiki kinerja produk Z, yaitu mengenai kesediaan seseorang membeli produk Z.

Tujuan penelitian ini adalah mengetahui ukuran *random forest* dan ukuran contoh peubah penjelas yang menghasilkan *random forest* berakurasi prediksi tinggi dan stabil, serta yang menghasilkan *driver analysis* yang stabil.

## TINJAUAN PUSTAKA

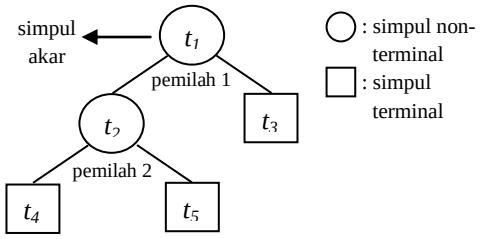
### *Driver Analysis*

*Driver analysis* merupakan istilah yang digunakan secara luas meliputi berbagai metode analisis. *Driver analysis* dilakukan untuk memahami pengaruh peubah penjelas terhadap peubah respons sehingga dapat diketahui prioritas setiap peubah penjelas dalam menggerakkan peubah respons (Weiner & Tang 2005). Metode

analisis yang digunakan dalam *driver analysis* disesuaikan dengan kondisi data yang dianalisis (Sambandam 2001).

***Classification and Regression Tree* (CART)**

CART merupakan metode eksplorasi data yang didasarkan pada teknik pohon keputusan. Pohon klasifikasi dihasilkan saat peubah respons berupa data kategorik, sedangkan pohon regresi dihasilkan saat peubah respons berupa data numerik (Breiman *et al*. 1984). Pohon terbentuk dari proses pemilahan rekursif biner pada suatu gugus data sehingga nilai peubah respons pada setiap gugus data hasil pemilahan akan lebih homogen (Breiman *et al.* 1984; Sartono & Syafitri 2010).




Gambar 1 Struktur Pohon pada Metode CART.

Pohon diilustrasikan dalam Gambar 1. Pohon disusun oleh simpul $t_1$, $t_2$, ..., $t_5$ (Gambar 1). Setiap pemilah (*split*) memilah simpul non-terminal menjadi dua simpul yang saling lepas. Hasil prediksi respons suatu amatan terdapat pada simpul terminal.

Menurut Breiman *et al.* (1984), pembangunan pohon klasifikasi CART meliputi tiga hal, yaitu:
1. Pemilihan pemilah (*split*)
2. Penentuan simpul terminal
3. Penandaan label kelas

***Random Forest***

Metode *random forest* adalah pengembangan dari metode CART, yaitu dengan menerapkan metode *bootstrap aggregating* (*bagging*) dan *random feature selection* (Breiman 2001). Dalam *random forest*, banyak pohon ditumbuhkan sehingga terbentuk hutan (*forest*), kemudian analisis dilakukan pada kumpulan pohon tersebut. Pada gugus data yang terdiri atas *n* amatan dan *p* peubah penjelas, *random forest* dilakukan dengan cara (Breiman 2001; Breiman & Cutler 2003):
1. Lakukan penarikan contoh acak berukuran *n* dengan pemulihan pada gugus data. Tahapan ini merupakan tahapan *bootstrap*.
2. Dengan menggunakan contoh *bootstrap*, pohon dibangun sampai mencapai ukuran maksimum (tanpa pemangkasan). Pada setiap simpul, pemilihan pemilah dilakukan dengan memilih *m* peubah penjelas secara acak, dimana $m << p$. Pemilah terbaik dipilih dari *m* peubah penjelas tersebut. Tahapan ini adalah tahapan *random feature selection*.
3. Ulangi langkah 1 dan 2 sebanyak *k* kali, sehingga terbentuk sebuah hutan yang terdiri atas *k* pohon.

Respons suatu amatan diprediksi dengan menggabungkan (*aggregating*) hasil prediksi *k* pohon. Pada masalah klasifikasi dilakukan berdasarkan *majority vote* (suara terbanyak).

Error klasifikasi *random forest* diduga melalui error OOB yang diperoleh dengan cara (Breiman 2001; Breiman & Cutler 2003; Liaw & Wiener 2002):
1. Lakukan prediksi terhadap setiap data OOB pada pohon yang bersesuaian. Data OOB (*out of bag*) adalah data yang tidak termuat dalam contoh *bootstrap*.
2. Secara rata-rata, setiap amatan gugus data asli akan menjadi data OOB sebanyak sekitar 36% dari banyak pohon. Oleh karena itu, pada langkah 1, masing-masing amatan gugus data asli mengalami prediksi sebanyak sekitar sepertiga kali dari banyaknya pohon. Jika *a* adalah sebuah amatan dari gugus data asli, maka hasil prediksi *random forest* terhadap *a* adalah gabungan dari hasil prediksi setiap kali *a* menjadi data OOB.
3. Error OOB dihitung dari proporsi misklasifikasi hasil prediksi *random forest* dari seluruh amatan gugus data asli.

Breiman dan Cutler (2003) menyarankan untuk mengamati error OOB saat dan *k* kecil, lalu memilih *m* yang menghasilkan error OOB terkecil. Jika *random forest* dilakukan dengan menghasilkan *variable importance*, disarankan untuk menggunakan banyak pohon, misalnya 1000 pohon atau lebih. Jika peubah penjelas yang dianalisis sangat banyak, nilai tersebut dapat lebih besar agar *variable importance* yang dihasilkan semakin stabil.

***Mean Decrease Gini***

*Mean Decrease Gini* (MDG) merupakan salah satu ukuran tingkat kepentingan (*variable importance*) peubah penjelas yang dihasilkan oleh metode *random forest*. Misalkan terdapat *p* peubah penjelas dengan , maka MDG mengukur tingkat kepentingan peubah penjelas $X_h$ dengan cara (Breiman & Cutler 2003; Sandri & Zuccolotto 2006):

dengan

: besar penurunan indeks Gini untuk peubah penjelas $X_h$ pada simpul *t*

*k* : banyaknya pohon dalam *random forest* (ukuran *random forest*)

## METODOLOGI

### Data

Data yang digunakan dalam penelitian ini adalah data sekunder yang diperoleh dari sebuah perusahaan riset pemasaran di Indonesia. Data tersebut terdiri atas sejumlah merek yang berbeda, dimana merek-merek tersebut merupakan jenis produk yang sama, yaitu produk Z. Banyaknya amatan dalam data adalah 1200 amatan.

Data yang digunakan terdiri atas sebuah peubah respons dan dua puluh peubah penjelas. Seluruhnya berskala pengukuran ordinal dengan lima kategori. Peubah responsnya adalah status kesediaan seseorang untuk membeli produk Z, sedangkan peubah penjelasnya adalah status kesetujuan seseorang terhadap atribut produk Z. Kategori masing-masing peubah dapat dilihat pada Tabel 1. Untuk melakukan metode *random forest* pada masalah klasifikasi, skala pengukuran data dianggap nominal.

Tabel 1 Kategori peubah penjelas dan peubah respons

| Peubah | Kategori peubah | |
|---|---|---|
| | Kode | Keterangan |
| Penjelas (X) | 1 | Sangat tidak setuju |
| | 2 | Tidak setuju |
| | 3 | Biasa saja |
| | 4 | Setuju |
| | 5 | Sangat setuju |
| Respons (Y) | 1 | Pasti tidak akan membeli |
| | 2 | Tidak akan membeli |
| | 3 | Tidak yakin akan membeli atau tidak |
| | 4 | Akan membeli |
| | 5 | Pasti akan membeli |

### Metode

1. Melakukan analisis statistika deskriptif terhadap peubah respons.
2. Melakukan simulasi *random forest.*
   a. Sebanyak 1000 *random forest* dibentuk pada setiap *m* dan *k* yang dicobakan, kemudian dicatat tingkat misklasifikasi masing-masing *random forest* dan *mean decrease gini* (MDG) setiap peubah penjelas. Nilai *m* dan *k* yang disarankan Breiman (2001) dicobakan dalam simulasi ini. Nilai *k* yang disarankan untuk digunakan pada metode *bagging* juga dicobakan, yaitu $k = 50$. Umumnya $k = 50$ sudah memberikan hasil yang memuaskan untuk masalah klasifikasi (Breiman 1996). Sementara itu, $k \geq 100$ cenderung menghasilkan tingkat misklasifikasi yang konstan (Sutton 2005). Nilai *m* dan *k* yang dicobakan adalah:

   dimana *p* adalah banyaknya peubah penjelas dalam data, yaitu $p = 20$.
   b. Menganalisis tingkat misklasifikasi *random forest* yang dihasilkan dari langkah 2a. Analisis dilakukan secara eksploratif.
   c. Melakukan *driver analysis* dengan metode *random forest*, yaitu mengamati urutan MDG peubah penjelas. MDG setiap peubah penjelas dihasilkan pada langkah 2a.
3. Melakukan analisis korelasi Spearman terhadap data.
4. Melakukan intepretasi hasil *driver analysis*.

Metode *random forest* dihasilkan menggunakan *software R ver* 2.12.0 dengan paket *randomForest ver* 3.6-2. Kriteria berhenti memilah yang digunakan adalah terdapatnya satu amatan pada simpul terminal.

## HASIL DAN PEMBAHASAN

### Analisis Deskriptif

Berdasarkan Tabel 2, diketahui terdapat 5 kategori pada peubah respons. Karena tidak ada responden yang menyatakan ‘pasti tidak akan membeli’, maka peubah respons yang dianalisis hanya terdiri atas 4 kategori. Dari 1200 responden, 56% responden menyatakan akan membeli produk Z, 41.7% responden menyatakan pasti akan membeli produk Z, 2% responden menyatakan tidak yakin akan membeli produk Z atau tidak membelinya, dan 0.3% responden menyatakan tidak akan membelinya. Secara deskriptif dapat dikatakan bahwa sebagian besar responden bersedia membeli produk Z.

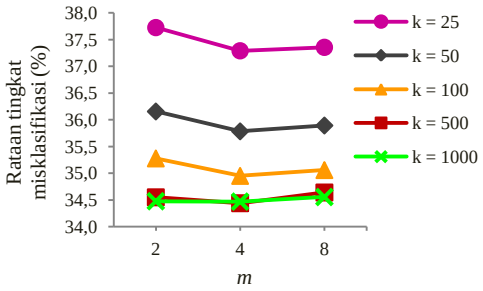
Tabel 2 Frekuensi dan persentase kategori peubah respons (status kesediaan seseorang untuk membeli produk Z)

| Kategori peubah respons | | Frekuensi | Persentase (%) |
|---|---|---|---|
| Kode | Keterangan | | |
| 1 | Pasti tidak akan membeli | 0 | 0.0 |
| 2 | Tidak akan membeli | 4 | 0.3 |
| 3 | Tidak yakin akan membeli atau tidak | 24 | 2.0 |
| 4 | Akan membeli | 672 | 56.0 |
| 5 | Pasti akan membeli | 500 | 41.7 |
| Total | | 1200 | 100.0 |

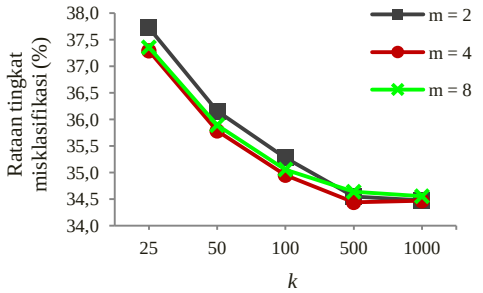
### Simulasi Ukuran *Random Forest* dan Ukuran Contoh Peubah Penjelas terhadap Keakuratan Prediksi *Random Forest*

Perubahan rataan tingkat misklasifikasi *random forest* akibat perubahan *m* disajikan dalam Gambar 2. Semakin besar *k*, perubahan

rataan tingkat misklasifikasi akibat perubahan *m* menjadi semakin tidak terlihat. Namun terlihat bahwa rataan tingkat misklasifikasi terendah selalu dicapai saat $\overline{\phantom{x}}$, yaitu *m* = 4, pada setiap *k* yang dicobakan. Ini menunjukan bahwa *m* = 4 adalah *m* optimal. Hal tersebut juga menunjukkan bahwa *m* optimal sudah dapat diketahui meski dengan *k* kecil. Dengan *m* = 4, *random forest* yang terbentuk merupakan *random forest* dengan pohon yang kuat, namun korelasi antar pohon cukup kecil.



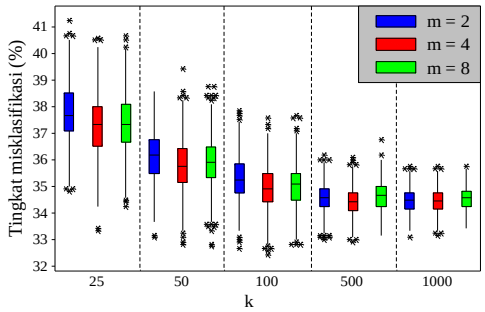

Gambar 2 Rataan tingkat misklasifikasi *random forest* berukuran *k* pada beberapa ukuran contoh peubah penjelas (*m*).




Gambar 3 Rataan tingkat misklasifikasi *random forest* berukuran contoh peubah penjelas *m* pada beberapa ukuran *random forest* (*k*).

Gambar 3 memperlihatkan perubahan rataan tingkat misklasifikasi akibat berubahnya *k*. Terlihat bahwa semakin besar *k* maka semakin kecil rataan tingkat misklasifikasi. Breiman (2001) menyatakan bahwa tingkat misklasifikasi *random forest* akan konvergen menuju nilai tertentu saat ukuran *random forest* semakin besar. Hasil simulasi (Gambar 3) sesuai dengan hal tersebut, yaitu ditunjukkan dengan saat *k* semakin besar, besarnya penurunan rataan tingkat misklasifikasi menjadi semakin tidak terlihat. Peningkatan *k* dari 500 pohon menjadi 1000 pohon terlihat tidak memberikan penurunan rataan tingkat misklasifikasi yang berarti. Dengan demikian, dapat dikatakan bahwa tingkat misklasifikasi *random forest* dalam memprediksi kesediaan membeli mulai konvergen saat menggunakan 500 pohon dan konvergen menuju 34.5%. Nilai tersebut adalah tingkat misklasifikasi terendah.

Penyebaran tingkat misklasifikasi menggambarkan kestabilan tingkat misklasifikasi. Dengan membandingkan seluruh diagram kotak garis pada Gambar 4, terlihat bahwa panjang diagram kotak garis cenderung konstan meskipun terjadi perubahan *m*. Akan tetapi, diagram kotak garis semakin memendek saat *k* meningkat. Ini menunjukkan bahwa kestabilan tingkat misklasifikasi *random forest* hanya bergantung pada *k*. Semakin besar *k* maka semakin stabil tingkat misklasifikasi *random forest*.




Gambar 4 Diagram kotak garis tingkat misklasifikasi *random forest* pada ukuran contoh peubah penjelas (*m*) dan ukuran *random forest* (*k*).

Gambar 4 juga memperlihatkan terdapatnya konvergensi tingkat misklasifikasi. Memendeknya diagram kotak garis terjadi secara perlahan dan bergerak menuju nilai tertentu. Saat *k* sebesar 1000, tingkat misklasifikasi *random forest* berada antara 33% dan 35.5%, dengan letak pemusatan terdapat pada nilai sekitar 34.5%. Pada *k* tersebut, kestabilan akurasinya adalah yang terbaik dibandingkan dengan pada *k* yang lebih kecil. Selain itu, letak pemusatannya merupakan nilai konvergensi tingkat misklasifikasi, juga merupakan tingkat misklasifikasi terendah yang dapat dicapai.

**Simulasi Ukuran *Random Forest* dan Ukuran Contoh Peubah Penjelas terhadap Hasil *Driver Analysis***

Pada penerapan *random forest* dalam *driver analysis* (DA-RF), *random forest* menghasilkan nilai *mean decrease gini* (MDG) untuk setiap peubah penjelas. *Driver analysis* dilakukan dengan memeringkatkan peubah penjelas berdasarkan MDG. Oleh karena itu, kestabilan MDG sangat menentukan kestabilan hasil *driver analysis*.

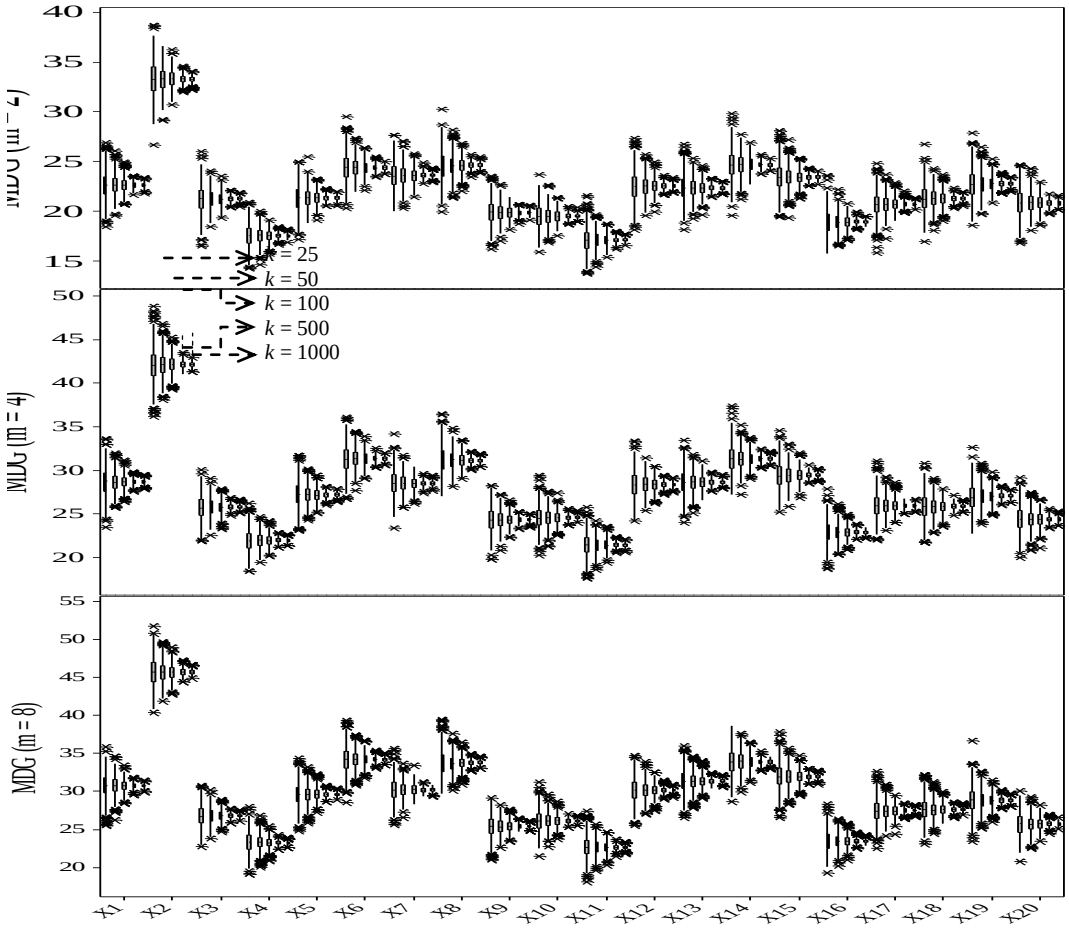
Hasil simulasi berupa diagram kotak garis MDG disajikan dalam Gambar 5. Tampak bahwa semakin besar *m* maka semakin besar nilai MDG. Akan tetapi, hal tersebut tidak mengubah panjang diagram kotak garis. Hasil ini menunjukkan bahwa keragaman MDG selalu sama besar pada *m*

berapapun, yang berarti *m* tidak mempengaruhi kestabilan MDG sehingga *m* tidak mengubah hasil *driver analysis*. Dengan demikian, diketahui bahwa *m* tidak mempengaruhi kestabilan hasil *driver analysis*.

Mengenai pengaruh *k* terhadap MDG, peningkatan *k* menyebabkan diagram kotak garis semakin pendek, yang berarti semakin besar *k* maka semakin stabil MDG. Berbeda dengan susunan diagram kotak garis pada Gambar 4, Gambar 5 memperlihatkan bahwa memendeknya diagram kotak garis tidak disertai dengan perubahan letak pemusatan MDG. Hasil tersebut menunjukkan bahwa kestabilan MDG sangat bergantung pada *k*, namun *k* tidak mempengaruhi besar perolehan MDG. MDG memiliki kestabilan yang baik saat *k* bernilai lebih dari 500, sehingga hasil *driver analysis* stabil pada *k* tersebut.

Telah diketahui bahwa *m* tidak mengubah hasil *driver analysis*, namun *random forest* memiliki akurasi tertinggi saat *m* = 4. Oleh karena itu, pengamatan hasil *driver analysis* selanjutnya dilakukan pada *driver analysis* saat *m* = 4. Hal tersebut dilakukan dengan menyusun *driver analysis* berdasarkan rataan MDG dari 1000 *random forest*. Hasilnya ditampilkan dalam Gambar 6. Seperti hasil sebelumnya, Gambar 6 juga memperlihatkan bahwa perubahan *k* tidak menyebabkan perubahan letak pemusatan, sehingga berapapun *k* yang digunakan tidak mempengaruhi rataan MDG peubah penjelas. Oleh sebab itu, penyusunan *driver analysis* berdasarkan rataan MDG menghasilkan *driver analysis* yang stabil.

Berdasarkan nilai rataaan MDG pada Gambar 6, terlihat bahwa hasil *driver analysis* pada *k* = 25 dan *k* = 50 sedikit berbeda dengan hasil *driver analysis* pada *k* lainnya (*k* = 100, 500, 1000). Pada *k* = 25, hal tersebut terjadi saat urutan X6-X14, yaitu dengan masing-masing nilai rataan MDG sebesar 31.319 dan 31.328. Sementara itu, pada *k* = 50, hal tersebut terjadi saat urutan $X_1$-$X_{13}$, dengan masing-masing nilai rataan MDG sebesar 28.651 dan 28.668. Karena nilai-nilai tersebut tidak terlalu berbeda jauh, maka hasil *driver analysis* berdasarkan rataan MDG tetap dapat dikatakan stabil meskipun menggunakan *k* yang bernilai kecil.




Gambar 5 Diagram kotak garis *mean decrease gini* (MDG) pada *random forest* (*m* = 2, 4, 8; *k* = 25, 50,100, 500, 1000).

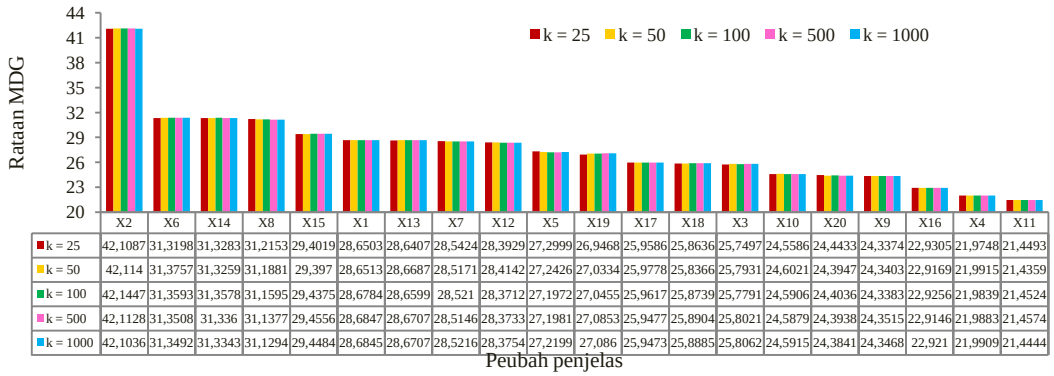
Dalam Gambar 6 diperlihatkan bahwa rataan MDG tertinggi dimiliki oleh $X_2$. Penurunan rataan MDG yang cukup drastis hanya terjadi pada peubah penjelas peringkat 1 dan 2, yaitu $X_2$ dan $X_6$. Pada peringkat selanjutnya, rataan MDG menurun secara lambat. Hal tersebut menunjukkan bahwa $X_2$ teridentifikasi sebagai atribut yang paling penting dalam mempengaruhi kesediaan membeli produk Z, serta memiliki pengaruh yang jauh lebih besar daripada pengaruh atribut lainnya. Ini menunjukkan bahwa memperbaiki atribut $X_2$ jauh lebih berpengaruh terhadap perbaikan kesediaan membeli dibandingkan dengan jika memperbaiki atribut lainnya. Oleh karena itu, untuk memperbaiki hal kesediaan seseorang dalam membeli produk Z, sangat diprioritaskan untuk memperbaiki atribut $X_2$. Prioritas berikutnya disesuaikan dengan hasil *driver analysis*. Urutan prioritas atribut berdasarkan hasil *driver analysis* adalah $X_2$-$X_6$-$X_{14}$-$X_8$-$X_{15}$-$X_1$-$X_{13}$-$X_7$-$X_{12}$-$X_5$-$X_{19}$-$X_{17}$-$X_{18}$-$X_3$-$X_{10}$-$X_{20}$-$X_9$-$X_{16}$-$X_4$-$X_{11}$.

Nilai koefisien korelasi Spearman antara kesediaan membeli produk Z dan atribut produk Z disajikan dalam Tabel 3. Arah koefisien korelasi Spearman menggambarkan bentuk hubungan antara suatu atribut dengan kesediaan seseorang membeli produk Z. Saat koefisien korelasi Spearman bernilai positif, maka diindikasikan bahwa terdapatnya suatu atribut di dalam produk Z mampu menggerakkan seseorang untuk bersedia membeli produk Z. Sebaliknya, koefisien korelasi Spearman yang bernilai negatif mengindikasikan bahwa tidak terdapatnya suatu atribut di dalam produk Z akan menggerakkan seseorang untuk bersedia membeli produk Z. Untuk atribut $X_2$, koefisien korelasi Spearman antara atribut $X_2$ dengan kesediaan membeli produk Z bernilai positif dan nyata pada taraf nyata 5%. Hasil ini menunjukkan bahwa terdapatnya atribut $X_2$ di dalam produk Z dapat menggerakkan seseorang untuk bersedia membeli produk Z.

Jika frekuensi terpilihnya suatu peubah penjelas untuk menjadi pemilah simpul dalam sebuah *random forest* diamati, maka terlihat bahwa atribut $X_2$ merupakan peubah penjelas yang paling sering terpilih sebagai pemilah simpul. Hal tersebut sejalan dengan hasil *driver analysis* berdasarkan rataan MDG. Akan tetapi, saat $m$ = 8, hal tersebut tampak tidak sejalan dengan hasil *driver analysis*. Saat $m$ = 8, atribut $X_6$ menjadi peubah penjelas yang paling sering terpilih sebagai pemilah simpul. Ini dapat terjadi karena untuk menghasilkan nilai MDG suatu peubah penjelas, nilai penurunan *impurity* peubah penjelas tersebut juga turut diperhitungkan. Nilai modus mengenai frekuensi terpilihnya suatu peubah penjelas untuk menjadi pemilah dalam sebuah *random forest* pada masing-masing $m$ dan $k$ yang dicobakan dapat dilihat pada Lampiran 1, Lampiran 2, dan Lampiran 3.

Tabel 3 Koefisien korelasi Spearman antara peubah penjelas dan peubah respons

| Peubah Penjelas | Korelasi | Nilai-p |
|---|---|---|
| $X_1$ | 0.091 | 0.002 |
| $X_2$ | 0.229 | 0.000 |
| $X_3$ | 0.159 | 0.000 |
| $X_4$ | 0.129 | 0.000 |
| $X_5$ | 0.138 | 0.000 |
| $X_6$ | 0.147 | 0.000 |
| $X_7$ | 0.224 | 0.000 |
| $X_8$ | 0.191 | 0.000 |
| $X_9$ | 0.143 | 0.000 |
| $X_{10}$ | 0.114 | 0.000 |
| $X_{11}$ | 0.146 | 0.000 |
| $X_{12}$ | 0.071 | 0.013 |
| $X_{13}$ | 0.149 | 0.000 |
| $X_{14}$ | 0.040 | 0.161 |
| $X_{15}$ | -0.013 | 0.659 |
| $X_{16}$ | 0.061 | 0.034 |
| $X_{17}$ | 0.071 | 0.014 |
| $X_{18}$ | 0.205 | 0.000 |
| $X_{19}$ | 0.237 | 0.000 |
| $X_{20}$ | 0.223 | 0.000 |




| | X2 | X6 | X14 | X8 | X15 | X1 | X13 | X7 | X12 | X5 | X19 | X17 | X18 | X3 | X10 | X20 | X9 | X16 | X4 | X11 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| k = 25 | 42,1087 | 31,3198 | 31,3283 | 31,2153 | 29,4019 | 28,6503 | 28,6407 | 28,5424 | 28,3929 | 27,2999 | 26,9468 | 25,9586 | 25,8636 | 25,7497 | 24,5586 | 24,4433 | 24,3374 | 22,9305 | 21,9748 | 21,4493 |
| k = 50 | 42,114 | 31,3757 | 31,3259 | 31,1881 | 29,397 | 28,6513 | 28,6687 | 28,5171 | 28,4142 | 27,2426 | 27,0334 | 25,9778 | 25,8366 | 25,7931 | 24,6021 | 24,3947 | 24,3403 | 22,9169 | 21,9915 | 21,4359 |
| k = 100 | 42,1447 | 31,3593 | 31,3578 | 31,1595 | 29,4375 | 28,6784 | 28,6599 | 28,521 | 28,3712 | 27,1972 | 27,0455 | 25,9617 | 25,8739 | 25,7791 | 24,5906 | 24,4036 | 24,3383 | 22,9256 | 21,9839 | 21,4524 |
| k = 500 | 42,1128 | 31,3508 | 31,336 | 31,1377 | 29,4556 | 28,6847 | 28,6707 | 28,5146 | 28,3733 | 27,1981 | 27,0853 | 25,9477 | 25,8904 | 25,8021 | 24,5879 | 24,3938 | 24,3515 | 22,9146 | 21,9883 | 21,4574 |
| k = 1000 | 42,1036 | 31,3492 | 31,3343 | 31,1294 | 29,4484 | 28,6845 | 28,6707 | 28,5216 | 28,3754 | 27,2199 | 27,086 | 25,9473 | 25,8885 | 25,8062 | 24,5915 | 24,3841 | 24,3468 | 22,921 | 21,9909 | 21,4444 |

Gambar 6 Urutan rataan *mean decrease gini* (MDG) pada *random forest* ($m$ = 4; $k$ = 25, 50,100, 500, 1000).

## SIMPULAN DAN SARAN

### Simpulan

*Random forest* berukuran contoh peubah penjelas sebesar 4 dan ukuran *random forest* lebih dari 500 memberikan akurasi prediksi yang tinggi dan stabil, yaitu dengan tingkat misklasifikasi berkisar antara 33% dan 35.5% dengan nilai rataannya sebesar 34.5%. Pada penerapan *random forest*, penyusunan *driver analysis* berdasarkan MDG menghasilkan *driver analysis* yang stabil jika ukuran *random forest* lebih dari 500. Untuk penyusunan *driver analysis* berdasarkan rataan MDG dari 1000 *random forest*, *driver analysis* tetap stabil meskipun menggunakan ukuran *random forest* cukup kecil. Hasil *driver analysis* pun stabil pada berbagai ukuran contoh peubah penjelas.

### Saran

Penelitian ini dilakukan pada ukuran *bootstrap* yang sama besar dengan ukuran data, yaitu sebesar 1200. Selain itu, juga dilakukan pada ukuran iterasi simulasi (banyaknya *random forest* dalam satu iterasi simulasi) sebesar 1000. Berkenaan dengan hal tersebut, saran untuk penelitian selanjutnya adalah:

1. Mengurangi ukuran *bootstrap* untuk melihat bagaimana pengaruhnya terhadap akurasi *random forest* dan hasil *driver analysis*. Salah satu keunggulan metode *random forest* adalah mampu menganalisis data yang ukuran datanya jauh lebih sedikit dibandingkan ukuran peubah penjelas dalam data (Breiman & Cutler 2001; Díaz-Uriarte & Andrés 2006).
2. Mengurangi ukuran iterasi simulasi untuk mengetahui ukuran iterasi yang efisien dalam menghasilkan *driver analysis* yang stabil.

## DAFTAR PUSTAKA


Brieman L, Friedman JH, Olshen RA, Stone CJ. 1984. *Classification and Regression Trees*. New York: Chapman & Hall.

Breiman L. 1996. Bagging Predictors. *Machine Learning* 24:123-140.

Breiman L. 2001. Random Forests. *Machine Learning* 45:5-32.

Breiman L, Cutler A. 2001. Random Forest. [terhubung berkala]. http://www.stat.berkeley.edu/~breiman/RandomForests/cc_home.htm#intro [8 Jul 2010].

Breiman L, Cutler A. 2003. Manual on Setting Up, Using, and Understanding Random Forest V4.0. [terhubung berkala]. http://oz.berkeley.edu/users/breiman/Using_random_forests_v4.0.pdf [8 Jul 2010].

Díaz-Uriarte R, Andrés SA de. 2006. Gene Selection and Classification of Microarray Data Using Random Forest. *BMC Bioinformatics* 7:3.

Liaw A, Wiener M. Des 2002. Classification and Regression by randomForest. *RNews Vol. 2/3*:18-22.

Sambandam R. 2001. Survey of analysis methods - Part I: key driver analysis. [terhubung berkala]. http://www.trchome.com/white-paper-library/wpl-all-white-papers/206 [30 Nop 2009].

Sandri M, Zuccolotto P. 2006. Variable Selection Using Random Forest. Di dalam: Zani S, Cerioli A, Riani M, Vichi M, editor. *Data Analysis, Classification and the Forward Search*; University of Parma, 6-8 Jun 2005. New York: Springer. hlm 263-270.

Sartono B, Syafitri UD. 2010. Ensemble Tree: an Alternative toward Simple Classification & Regression Tree. *Forum Statistika dan Komputasi* 15(1):1-7.

Sutton CD. 2005. Classification and Regression Trees, Bagging, and Boosting. *Handbook of Statistics* 24:303-329.

Wiener JL, Tang J. 2005. Multicollinearity in Customer Satisfaction Research. Ipsos Loyalty.

Lampiran 1 Modus frekuensi terpilihnya peubah penjelas sebagai pemilah (*split*) simpul dalam sebuah *random forest* dengan ukuran contoh peubah penjelas (*m*) sebesar 2 dan ukuran *random forest* (*k*) sebesar 25, 50, 100, 500, dan 1000

| Peubah Penjelas | Ukuran *Random Forest* (*k*) | | | | |
|---|---|---|---|---|---|
| | 25 | 50 | 100 | 500 | 1000 |
| $X_1$ | 388 | 789 | 1564 | 7817 | 15523 |
| $X_2$ | 421 | 856 | 1731 | 8700 | 17380 |
| $X_3$ | 350 | 712 | 1446 | 7215 | 14289 |
| $X_4$ | 342 | 715 | 1405 | 6972 | 13985 |
| $X_5$ | 374 | 778 | 1523 | 7480 | 14908 |
| $X_6$ | 424 | 834 | 1646 | 8310 | 16624 |
| $X_7$ | 356 | 691 | 1415 | 7040 | 14079 |
| $X_8$ | 389 | 786 | 1568 | 7790 | 15598 |
| $X_9$ | 357 | 719 | 1434 | 7076 | 14106 |
| $X_{10}$ | 376 | 730 | 1493 | 7482 | 14894 |
| $X_{11}$ | 320 | 682 | 1334 | 6678 | 13095 |
| $X_{12}$ | 376 | 763 | 1549 | 7781 | 15596 |
| $X_{13}$ | 372 | 748 | 1517 | 7657 | 15201 |
| $X_{14}$ | 411 | 829 | 1606 | 8155 | 16271 |
| $X_{15}$ | 394 | 768 | 1555 | 7749 | 15486 |
| $X_{16}$ | 357 | 716 | 1440 | 7163 | 14456 |
| $X_{17}$ | 371 | 758 | 1527 | 7681 | 15341 |
| $X_{18}$ | 322 | 679 | 1354 | 6604 | 13344 |
| $X_{19}$ | 315 | 642 | 1303 | 6398 | 13016 |
| $X_{20}$ | 292 | 571 | 1136 | 5778 | 11461 |

Lampiran 2 Modus frekuensi terpilihnya peubah penjelas sebagai pemilah (*split*) simpul dalam sebuah *random forest* dengan ukuran contoh peubah penjelas (*m*) sebesar 4

| Peubah Penjelas | Ukuran *Random Forest* (*k*) | | | | |
|---|---|---|---|---|---|
| | 25 | 50 | 100 | 500 | 1000 |
| $X_1$ | 437 | 858 | 1768 | 8881 | 17677 |
| $X_2$ | 493 | 995 | 1994 | 10041 | 19990 |
| $X_3$ | 368 | 742 | 1493 | 7494 | 15041 |
| $X_4$ | 384 | 783 | 1566 | 7780 | 15479 |
| $X_5$ | 438 | 885 | 1720 | 8692 | 17337 |
| $X_6$ | 482 | 983 | 1976 | 9804 | 19671 |
| $X_7$ | 373 | 716 | 1443 | 7442 | 14828 |
| $X_8$ | 450 | 876 | 1819 | 8984 | 17914 |
| $X_9$ | 382 | 784 | 1550 | 7710 | 15411 |
| $X_{10}$ | 414 | 836 | 1669 | 8322 | 16771 |
| $X_{11}$ | 357 | 722 | 1455 | 7268 | 14383 |
| $X_{12}$ | 444 | 887 | 1789 | 8927 | 17846 |
| $X_{13}$ | 431 | 871 | 1749 | 8742 | 17476 |
| $X_{14}$ | 457 | 941 | 1838 | 9231 | 18315 |
| $X_{15}$ | 418 | 841 | 1681 | 8414 | 16899 |
| $X_{16}$ | 392 | 750 | 1526 | 7570 | 15180 |
| $X_{17}$ | 411 | 838 | 1691 | 8407 | 16826 |
| $X_{18}$ | 355 | 685 | 1407 | 7136 | 14139 |
| $X_{19}$ | 322 | 644 | 1309 | 6405 | 12866 |
| $X_{20}$ | 264 | 558 | 1083 | 5421 | 10888 |

Lampiran 3 Modus frekuensi terpilihnya peubah penjelas sebagai pemilah (*split*) simpul dalam sebuah *random forest* dengan ukuran contoh peubah penjelas (*m*) sebesar 8

| Peubah Penjelas | Ukuran *Random Forest* (*k*) | | | | |
|---|---|---|---|---|---|
| | 25 | 50 | 100 | 500 | 1000 |
| $X_1$ | 420 | 839 | 1663 | 8351 | 16707 |
| $X_2$ | 488 | 946 | 1935 | 9418 | 19072 |
| $X_3$ | 317 | 649 | 1296 | 6465 | 12987 |
| $X_4$ | 349 | 711 | 1437 | 7136 | 14306 |
| $X_5$ | 421 | 842 | 1662 | 8386 | 16751 |
| $X_6$ | 490 | 980 | 1954 | 9756 | 19578 |
| $X_7$ | 326 | 677 | 1327 | 6743 | 13439 |
| $X_8$ | 415 | 867 | 1739 | 8584 | 17150 |
| $X_9$ | 347 | 688 | 1388 | 6934 | 13879 |
| $X_{10}$ | 389 | 773 | 1556 | 7739 | 15546 |
| $X_{11}$ | 326 | 652 | 1316 | 6620 | 13260 |
| $X_{12}$ | 417 | 820 | 1641 | 8245 | 16590 |
| $X_{13}$ | 419 | 839 | 1703 | 8379 | 16885 |
| $X_{14}$ | 430 | 850 | 1712 | 8478 | 16870 |
| $X_{15}$ | 386 | 764 | 1529 | 7666 | 15272 |
| $X_{16}$ | 323 | 660 | 1314 | 6570 | 13130 |
| $X_{17}$ | 387 | 760 | 1530 | 7654 | 15425 |
| $X_{18}$ | 334 | 648 | 1315 | 6504 | 13071 |
| $X_{19}$ | 276 | 555 | 1074 | 5442 | 10982 |
| $X_{20}$ | 227 | 443 | 938 | 4630 | 9260 |